

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 05-081192

(43)Date of publication of application : 02.04.1993

(51)Int.Cl. G06F 13/42  
G06F 13/36

(21)Application number : 03-267110

(71)Applicant : FUJITSU LTD

(22)Date of filing : 18.09.1991

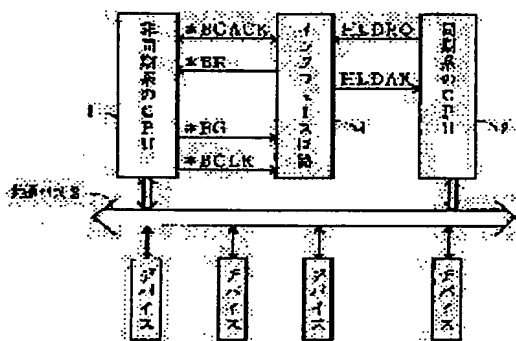
(72)Inventor : AYABE MASAKATSU

## (54) BUS ARBITRAION METHOD

## (57)Abstract:

PURPOSE: To connect different kinds of families with each other by providing the interface circuit having specific functions between asynchronous group CPUs and synchronous group CPUs.

CONSTITUTION: An interface circuit 4 is provided between an asynchronous group CPU 1 and a synchronous group CPU 2. An asynchronous group signal is converted into a synchronous group signal and a synchronous group signal into an asynchronous group signal, and then bus arbitration for a common bus 3 is performed. When the interface circuit 4 receives a bus use request signal HLDRQ from the CPU 2, it conveys a bus use request signal \*BR to the CPU 1. Responding to this, the CPU 1 asynchronously transmits a bus use authorization signal \*BG to the interface circuit 4. Then the interface circuit 4, responding to this, informs the CPU 2 of a bus acquisition signal HLDAK. At this time, the CPU 2 is informed of it through synchronous signal conversion.



## LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)特許出願公開番号

特開平5-81192

(43)公開日 平成5年(1993)4月2日

(51)Int.Cl.<sup>5</sup>

G 0 6 F 13/42  
13/36

識別記号

3 4 0 B 8840-5B  
5 2 0 D 7052-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数3(全 9 頁)

(21)出願番号 特願平3-267110

(22)出願日 平成3年(1991)9月18日

(71)出願人 000005223

富士通株式会社

神奈川県川崎市中原区上小田中1015番地

(72)発明者 綾部 正勝

神奈川県川崎市中原区上小田中1015番地

富士通株式会社内

(74)代理人 弁理士 茂泉 修司

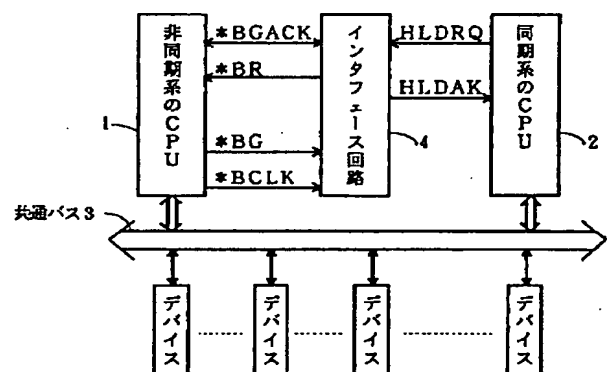
(54)【発明の名称】 バスアービトレーション方式

(57)【要約】

【目的】 二つのCPUが共通のバスに接続されているときのバスのアービトレーション(調停)方式に関し、CPUと異種ファミリのデバイスとの調停を行うことによって異種ファミリ間の接続を可能にすることを目的とする。

【構成】 非同期系のCPU1と同期系のCPU2との間にインタフェース回路4を設け非同期系の信号を同期系の信号に、また、同期系の信号を非同期系の信号に変換して共通バス3へのバスアービトレーションを行うように構成する。

本発明の原理図



## 【特許請求の範囲】

【請求項1】 非同期系のCPU(1)と同期系のCPU(2)とが共通のバス(3)に接続されているバスアービトレーション方式において、

非同期系CPU(1)が該同期系CPU(2)からバス使用要求信号(HLRQ)を受けたとき、該非同期系CPU(1)に対してバス使用要求信号(\*BR)を伝え、このバス使用要求信号(\*BR)によって該非同期系CPU(1)がバス使用許可信号(\*BG)を与えたときに該同期系CPU(2)に対してバス獲得信号(HLDAK)を該同期系CPU(2)に同期信号変換して通知すると共に該非同期系CPU(1)に対してバス獲得承認信号(\*BGACK)を与えて該同期系CPU(2)が該バス(3)を獲得したことを通知するインタフェース回路(4)を設けたことを特徴とするバスアービトレーション方式。

【請求項2】 該インタフェース回路(4)は、該同期系CPU(2)のバス使用要求信号(HLRQ)が終了したことを知らされたとき該バス獲得信号(HLDAK)及びバス獲得承認信号(\*BGACK)を無効にして該同期系CPU(2)のアクセスが終了したことを知らせることを特徴とした請求項1に記載のバスアービトレーション方式。

【請求項3】 該インタフェース回路(4)は、該非同期系CPU(1)からバス開放要求信号(\*BCLR)を受けたとき、該バス獲得信号(HLDAK)を該同期系CPU(2)に同期信号変換して無効にし該バス使用要求信号(HLRQ)を該同期系CPU(2)に同期信号変換して無効にすることにより該同期系CPU(2)をバス待ち状態にした後該同期系CPU(2)が所定の期間だけ該バス使用要求信号(HLRQ)を無効にした後再び有効にしたとき該バス獲得承認信号(\*BGACK)を無効にすることにより該非同期系CPU(1)のバス使用を可能にしことを特徴とした請求項1に記載のバスアービトレーション方式。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】本発明はバスアービトレーション方式に関し、特に二つのCPUが共通のバスに接続されているときのバスのアービトレーション(調停)方式に関するものである。

【0002】複数のCPUが共通のバスに接続されているときには、それぞれのCPUが交互にバスを使用するための調停を行い、どちらが使用するかを決める必要がある。

## 【0003】

【従来の技術】現在、CPUを有するシステムは、ほぼ二分されており同期系のもの(例えば米国インテル社のもの)、非同期系のもの(例えば米国モトローラ社のもの)があり、このようなCPUを使ってシステムを構築する場合には、始めにホストのCPUを決め、これに必要な機能を満たすスレーブCPU等のデバイスを接続して行く。

【0004】そして、上記の同期系の場合には、ホストCPUとスレーブCPUとの間で一定の同期クロックに基づいてどちらのCPUがバスを獲得するかを決定してバスを使用することになるが、後者の非同期系の場合には、特に同期クロックなどを使わずにホストCPUとスレーブCPUとの間で信号のやりとりを随時に行いバス獲得のための調停を行っている。

## 【0005】

【発明が解決しようとする課題】このようにバスのアービトレーションを行う場合、ホストCPUを一度決定すると、付加されるスレーブCPU等のデバイスも必然的にホストと同じ同期系又は非同期系のファミリに限定されてしまう。この理由は信号体系や信号機能が全く異なっているからである。

【0006】従って、現在においては異種ファミリ間のバスアービトレーションが出来ないという問題点があった。

【0007】そこで本発明は、CPUと異種ファミリのデバイスとの調停を行うことによって異種ファミリ間の接続を可能にしたバスアービトレーション方式を実現することを目的とする。

## 【0008】

【課題を解決するための手段及び作用】本発明に係るバスアービトレーション方式は、図1に概念的に示すように非同期系のCPU1と同期系のCPU2との間にインタフェース回路4を設け非同期系の信号を同期系の信号に、また、同期系の信号を非同期系の信号に変換して共通バス3へのバスアービトレーションを行うものである。

【0009】このような本発明に係るバスアービトレーション方式の動作を図2及び図3に示す動作タイムチャートにより以下説明する。尚、\*が付してある記号は非同期系の信号であることを示しており、それ以外は同期系の信号であることを示している。

【0010】先ず図2に示す方式については、非同期系CPU1がバス3を使用していないとき、即ちバス獲得承認信号\*BGACKが無効の時(非同期系ではネゲートと称される状態の時)、同期系CPU2からバス使用要求信号HLDRQが有効になってこの要求を受けた時(これは同期系では非同期系と反対の論理状態となるものでありイネーブル状態と称される時)、インタフェース回路4は非同期系CPU1に対してバス使用要求信号\*BRを有効状態にしてその要求を伝える。

【0011】このバス使用要求信号\*BRを受けた非同期系のCPU1ではこれに回答して非同期にバス使用許可信号\*BGを有効にしてバスの使用許可を与える旨インタフェース回路4に知らせる。

【0012】これを受けてインタフェース回路4では同期系CPU2に対しバス獲得信号HLDAKを通知する。但し、この時には同期系CPU2に対して同期信号

3

変換により通知を行う。

【0013】これと共にインタフェース回路4は非同期系CPU1に対してバス獲得承認信号\*BGACKを有効にして同期系CPU2がバス3を獲得した旨通知し、非同期系CPU1からのバス使用を禁止させる様にしている。この様にして異種ファミリ間のバスアービトレーションがインタフェース回路4の非同期系/同期系間の調停により行われる事となる。

【0014】尚、このインタフェース回路4は、同期系CPU2のバス使用が終了してバス使用要求信号HLD RQが無効になったことが通知された時、CPU2へのバス獲得信号HLDAK及びCPU1へのバス獲得承認信号\*BGACKをそれぞれ無効にして同期系CPU2のバス使用が終わった事を知らせる事が出来る。

【0015】次に図3に示した本発明によるバスアービトレーション方式をすると、同期系のCPU2が上記の様にバス使用権を獲得する迄は図2と同じであるが、このままであると非同期系のCPU1はCPU2の使用終了時まではバス3の使用が出来なくなってしまう。

【0016】そこでこの発明では非同期系のCPU1がバスの使用を行いたい時、同期系のCPU2のバス使用状態を強制的に中断させ、緊急度の高いサービスを行おうとするものである。

【0017】即ち、この様な強制中断を行う時には非同期系CPU1からバス開放要求信号\*BCLRが有効となり、これを受けたインタフェース回路4では、同期系CPU2へのバス獲得信号HLDAKを有効にさせる。尚、この無効にさせる動作も同期系CPU2に同期信号変換して行われる。

【0018】この様にバス獲得信号HLDAKが有効にされた同期系CPU2ではバス使用要求信号HLD RQを無効にした後、且つ図示の様に所定の期間だけ無効にしてから再び有効にする。

【0019】これによりインタフェース回路4はバス獲得承認信号\*BGACKを無効にする事により非同期系CPU1のバス使用を可能にしている。この時、上記の様に同期系CPU2はバス獲得信号HLDAKが無効になっている事によりバス待ち状態になっている。

【0020】

【実施例】図4は、本発明に係るバスアービトレーション方式の実施例を示したもので、この実施例では特に図2に示した強制中断機能が無い場合の実施例を示している。

【0021】図中、1は図1に示した非同期系のCPU1からの信号であるバス獲得承認信号\*BGACKと同期系CPU2からのバス使用要求信号HLD RQとを入力するANDゲート、2はANDゲート1の出力信号を反転して非同期系CPU1へのバス使用要求信号\*BRに信号変換する為のインバータ、3はCPU1からのアドレスストローブ信号\*ASを反転するインバータ、4

4

はCPU1からのバス使用許可信号\*BGを反転するインバータ、5は上記のバス使用要求信号HLD RQとシステムのリセット信号\*RSTとを入力するANDゲート、6は5ボルトの電源をデータ入力としインバータ4の出力信号をクロックとしてデータ入力を叩くと共にANDゲート5からの出力信号によりリセットされるフリップフロップ（以下、FFと言う）、7はインバータ3の出力信号とFF6の反転出力とを入力してCPU2へのバス獲得信号HLDAKを発生するNORゲート7、そして8はNORゲート7の出力信号を反転してCPU1へのバス獲得承認信号\*BGACKを与えるインバータである。尚インバータ8の出力信号\*BGACKは、そのままANDゲート1の入力信号にもなっている。

【0022】このような構成の実施例の動作を上記の図1及び図2を参照して説明する。尚、同期系CPUでの信号の有効/無効（イネーブル/ディセーブル）は、非同期系CPUでの信号の有効/無効と合わせるため、ここではアサート/ネゲートとそれぞれ称することとする。従って、両CPU間ではアサート/ネゲートは互いに反対の論理レベルとなる。

【0023】まず、CPU2が何らかの要因によりバス使用要求信号HLD RQを発生すると、このバス使用要求信号HLD RQは有効、即ちアサート（“H”）される。この時、CPU1がバス3を使用していなければ、即ちバス獲得承認信号\*BGACKがネゲート（“H”）されていれば、ANDゲート1は両入力が“H”になってその出力レベルも“H”となり、インバータ2を経てレベル“L”のアサートされたバス使用要求信号\*BRがCPU1に与えられる。

【0024】これによりCPU1はバス要求許可信号\*BGをアサート（“L”）し、これがインバータ4を介してFF6にクロックとして与えられることによりFF6の反転出力端子からNORゲート7への信号は“L”レベルとなりCPU1からのアドレスストローブ信号\*ASがネゲート（“L”）されるまで待つことにより、NORゲート7の出力信号、即ちCPU2へのバス獲得信号HLDAKがアサート（“H”）されてCPU2がバス3を獲得した旨知らされる。

【0025】尚、アドレスストローブ信号\*ASとは非同期系CPU1が自己のサイクルの終了を他のデバイスに通知するための信号であり、このアドレスストローブ信号\*ASが必要なのは、CPU1は、外部デバイスから所定のタイミングでの出力信号を受けないとバス使用権の要求が無かったものとしてアイドル・サイクルに戻ってしまうが、このバス獲得承認信号\*BGACKを受けると非同期系のためアドレスストローブ信号\*ASによるサイクルの終了を検知するまでアイドル・サイクルに戻らないようにするためである。

【0026】また、FF6でラッチするのは、上記の場合において、非同期信号であるバス使用許可信号\*BG

5

がアイドル・サイクルに戻る前にネゲート（“H”）されてしまうが、バス獲得信号HLD AKをネゲートされないまま（“H”）にして同期系CPU2に対するタイミング整合が必要であるからである。

【0027】そして、この様にしてバス獲得信号HLD AKがアサートされた時、インバータ8を介してバス獲得承認信号\*BGACKがアサート（“L”）され、CPU1に対して外部のCPU2がバスマスターとしてバス3を獲得した旨通知する事となる。

【0028】又、バス獲得承認信号\*BGACKがアサートされたことによりCPU2からのバス使用要求信号HLDRQはANDゲート1でマスクされる事となり、CPU1に対するリクエスト要求を消去している。即ち、CPU2からのバス使用要求信号HLDRQはネゲートさせないこととなる。これは同期信号を非同期信号に変換するためである。

【0029】現在のバスマスタであるCPU2のバス使用要因が終了し、バス使用要求信号HLDRQがネゲート（“L”）されると、ANDゲート5を介してFF6にリセットが掛かりNORゲート7を介してバス獲得信号HLD AKがネゲートされると共に、インバータ8を介してバス獲得承認信号\*BGACKもネゲートされることとなる。

【0030】これによってCPU1は外部アクセスが終了したことを認識し、図2に示すようにアイドル・サイクルへ移る。

【0031】尚、この実施例では、外部のバスマスタになりうるデバイスはCPU2のみであるが、非同期系の外部バスマスタになりうるデバイスを更に付加する場合は、CPU1に対して信号\*BR、\*BG、\*BGACKにデバイスの同じ入力を接続し、この回路の出力をオープンコレクタもしくはオープンドレインにすることで対応することができる。

【0032】上記の実施例では一度DMAC（直接メモリアクセス）がバスを獲得すると、サイクル終了迄はバスを開放しないのでサイクルスチルモード（バースト禁止）の様なアクセス方法が好ましい。

【0033】図5はバーストモードで大量のデータを適宜送出するような場合、アービトレーションを強制中断して緊急度の高いサービスを行う為の実施例を示したもので、この実施例は、図4の実施例に対して、NORゲート7の前にインバータ3の出力とFF6の反転出力とを入力するORゲート11と、バス開放要求信号\*BCLRを反転するインバータ12と、16MHzのクロックを入力しインバータ12の出力信号をデータ入力とするFF13と、バス使用要求信号HLDRQをクロック信号としインバータ12の出力信号をデータ入力とするFF14と、インバータ12の出力信号15とFF13の出力信号とを入力するORゲート15と、このORゲート15の出力信号とORゲート11の出力信号とを入力

6

するORゲート16と、このORゲート16の出力信号とバス開放要求信号\*BCLRとを入力するANDゲート17と、このANDゲート17の出力信号とFF14の出力信号とを入力してインバータ8へ与えるNORゲート18とを含んでおり、NORゲート7はORゲート11と15の出力を入力するようになっている。

【0034】このような強制中断機能を有する実施例の動作を図1及び図3を参照して以下に説明する。この実施例における動作は図2及び図3から判る様にバス使用要求信号HLDRQがCPU2から与えられてバス獲得承認信号\*BGACKがアサートされてCPU2のバス獲得を示すまでは上記の図4の実施例と同様である。

【0035】ここで、同期系と非同期系のタイミングマネージメントについて説明すると、同期系ではバスの獲得要求がCPUに通知されるとタイミングマネージメントを完了しCPUが許可を行う。この許可は要求が消去されるまで継続され、CPU内部で緊急のサービス等が発生したときにはその許可を取り消すことによりバスを取り戻すことができる。

【0036】一方、非同期系では、バス獲得要求がCPUに通知されると、CPUは内部で一定の処理時間後、バス権の要求許可信号を直ちに出力する。これは、CPUのサイクル中でも起こり得る。このため、許可信号を受け取った他のデバイス（スレーブ）側では、タイミングマネージメント（アドレスストローブ信号\*ASによるサイクル終了検知）を行ったあと、一定の時間を置き、バス獲得承認信号\*BGACKの出力とバス獲得要求信号の取り下げを行う。CPU側では、このとき、バス獲得要求信号が取り下げられないと他のデバイス（スレーブ）が未だ要求しているものと見て、一定時間アサートしていたバス使用許可信号\*BGをネゲートし、また、一定時間を置いてバス使用許可信号\*BGを再アサートする。他のデバイスはこの再アサートを検知し、タイミングマネージメントしてバスを獲得するものである。

【0037】このように同期系と非同期系とではタイミング整合の有無が一番大きな違いとなっており、両者間の調整が必要となる。

【0038】まず、バス開放要求信号\*BCLRがCPU1からアサート（“L”）されて与えられると、インバータ12を介してFF13の出力が“L”レベルとなり、インバータ12の出力信号をORゲート15が入力していることからNORゲート7の出力であるバス獲得信号HLD AKはネゲート（“L”）されてCPU2のバス待ち動作がトリガー（緊急割込）され、これによりCPU2はバス使用要求信号HLDRQが即座にネゲートされ、バス3を開放して、バス待ち動作に移る。

【0039】そして、このバス待ち動作に移ると、CPU2では予め決まっている2マスタクロックによりバス使用要求HLDRQを一旦ネゲートした後、再びバス使

7

用要求HLD RQをアサートし、これによりCPU2はバス待ち状態に完全に移行したこととなる。

【0040】この様に、再びアサートされたバス使用要求信号HLD RQによりFF14の出力信号が“L”→“H”に変化し、この結果、ORゲート18及びインバータ8を介してバス獲得承認信号\*BGACKをネゲートする。このように、バス3の使用状況を知らせる信号はバス獲得承認信号\*BGACKだけなので、このバス獲得承認信号\*BGACKのネゲート状態をバスが完全開放された状態になるまでFF14で待たせる必要がある。即ち、再度バス使用要求信号HLD RQがアサートされるまで非同期系CPU1はバス3が開放されていないものと認識していることとなる。このようにして同期信号から非同期信号への変換が行われる。

【0041】又、高位のCPU1によるサービスが終了すると、FF14が通常モードにセットされ、またFF13が16MHzクロックの次の立ち上がりで通常モードにセットされる。これにより、バス開放要求信号\*BCLRが再アサートされ続けている非同期系CPU2に対してバス獲得信号HLD AKを出力することが可能となる。但し、直ちにバス使用権を明け渡すとデータの競合が起こり得るので、FF13で16MHzクロックの立ち上がり分だけ保障している。

【0042】また、CPU1はバス開放要求信号\*BCLRがネゲートされるまでバス使用許可信号\*BGをアサートさせないため、FF13、14が通常モードに移っても直ちにはバス獲得信号HLD AKをアサートさせないようにしている。バス獲得信号HLD AKを検知するとCPU2は中断していたDMA転送を再開することとなる。この動作は高位のサービス要求が生じる毎に発生する。

【0043】

【発明の効果】以上の様に本発明に係るバスアービトレーション方式によれば、インタフェース回路を用いて非

8

同期系のCPUと同期系のCPUとの信号の整合変換を行っているので異種ファミリのCPU間の接続が可能となる。

【0044】即ち、これまでは使用環境条件が処理能力、アプリケーションの豊富さや信頼性等の観点からホストCPUが決定され、同系ファミリの中からアプリケーションを選定していたが、本発明により異種ファミリのインタフェースを用いることによって異種間ファミリの接続が容易に行えることとなった。

10 【図面の簡単な説明】

【図1】本発明に係るバスアービトレーション方式を原理的に示したブロック図である。

【図2】本発明に於ける強制中断が無いときの動作を示したタイムチャート図である。

【図3】本発明に於ける強制中断が有る時の動作を示したタイムチャート図である。

【図4】本発明に係るバスアービトレーション方式の実施例（強制中断無し）を示した回路図である。

20 【図5】本発明に係るバスアービトレーション方式の実施例（強制中断有り）を示した回路図である。

【符号の説明】

1 非同期系CPU

2 同期系CPU

3 共通バス

4 インタフェース回路

5 HLD RQ バス使用要求信号

6 HLD AK バス獲得信号

7 \*BR バス使用要求信号

8 \*BG バス使用許可信号

30 9 \*BGACK バス獲得承認信号

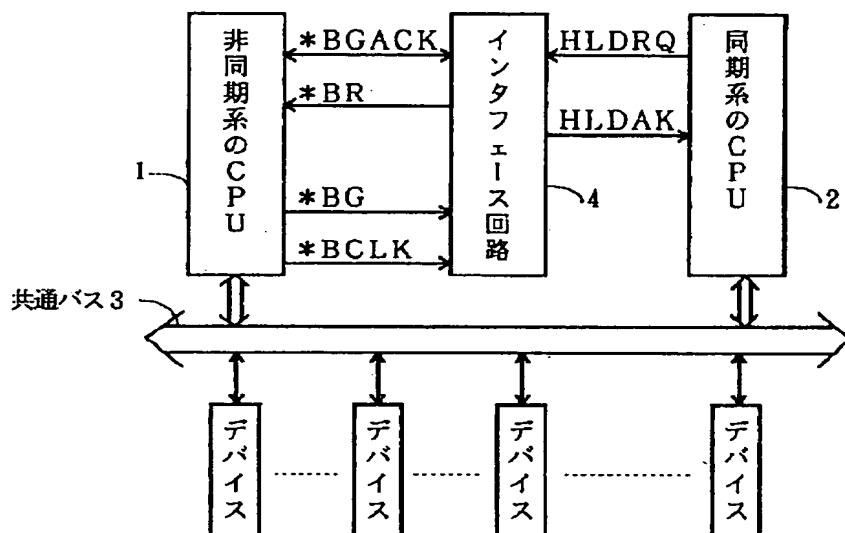
10 \*BCLR バス開放要求信号

11 \*AS アドレスストローブ信号

図中、同一符号は同一又は相当部分を示す。

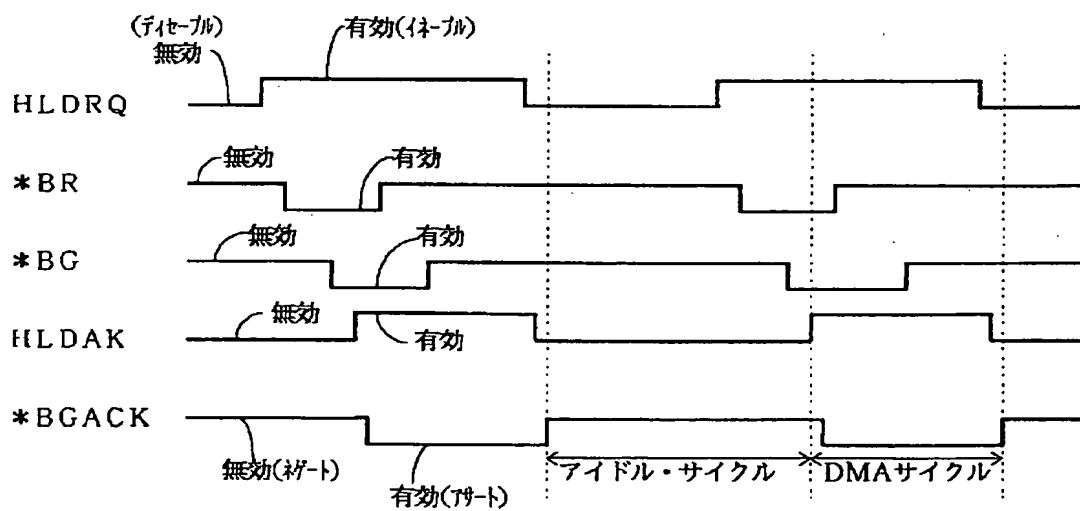
【図1】

## 本発明の原理図



【図2】

## 本発明の動作タイムチャート（強制中断無し）



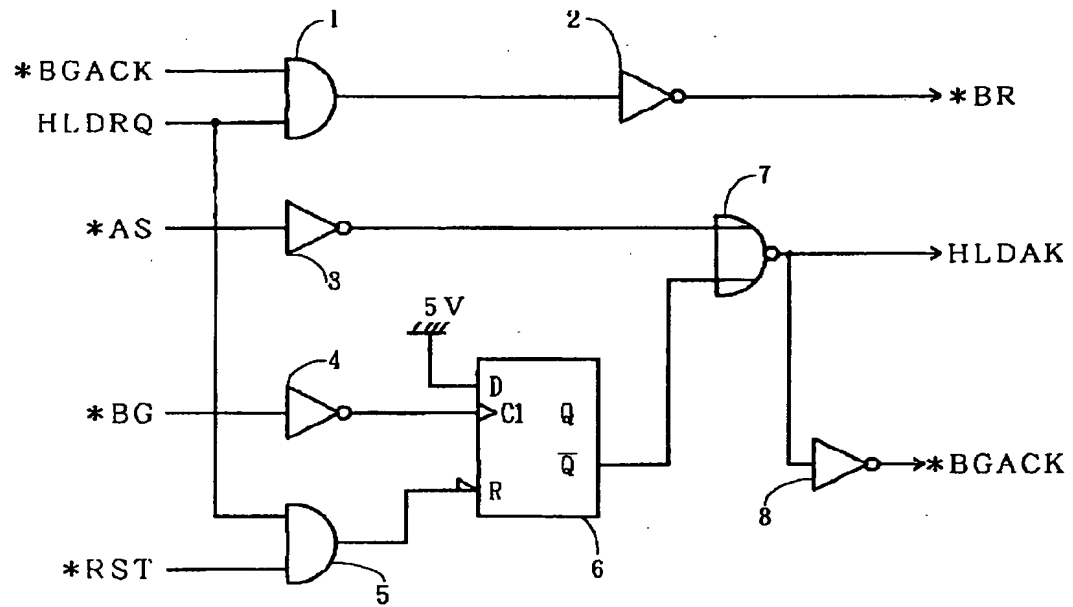


本発明の動作タイムチャート（強制中継有り）



【図4】

本発明の実施例(強制中断無し)



【図5】

本発明の実施例（強制中断有り）

